$$\sum_{q\in Q(m)} \pi_1(q) \le R_1(m),\ \gamma(q) \ge 0$$]

is solved using the approximation ~~in EQ. 11,~~ [

$$\gamma(q) = \frac{P_{21}(q,q)\pi_1(q)}{i_2(q)}$$]

and the network further comprises at least one high-level network controller that controls the power constraints $\cancel{R_1(q)}$ [$R_1(m)$], and drives the network towards a max-min solution.

73. (currently amended) A method as in claim ~~60~~[61], wherein each node:

is given an initial $\boldsymbol{\gamma}_0$;

generates the model expressed in ~~EQ. 20, EQ. 21, and EQ. 22~~ [

$$L(\boldsymbol{\gamma},\mathbf{g},\beta) = \mathbf{g}^T\boldsymbol{\gamma},\ \Sigma_{q\in Q(m)} \log(1+\gamma(q)) \ge \beta(m)$$

$$\mathbf{g} = \nabla_\gamma f(\boldsymbol{\gamma}_0)$$];

updates the new $\boldsymbol{\gamma}_\alpha$ from ~~EQ. 23 and EQ. 24~~

[$\boldsymbol{\gamma}_* = \arg\min_\gamma L(\boldsymbol{\gamma},\mathbf{g},\beta)$, $\boldsymbol{\gamma}_\alpha = \boldsymbol{\gamma}_0 + \alpha(\boldsymbol{\gamma}_* - \boldsymbol{\gamma}_0)$];

determines a target SINR to adapt to; and,

updates the transmit power for each link q according to ~~EQ. 25 and EQ. 26~~ [

$$\pi_2(q) = \gamma_\alpha i_1(q) \Big/ |h(q)|^2$$

$$\pi_1(q) = \gamma_\alpha i_2(q) \Big/ |h(q)|^2$$].

74. (currently amended) A method as in claim ~~60~~[61], for each node wherein the transmit power relationship of ~~EQ. 25 and EQ 26~~ [

$$\pi_2(q) = \gamma_\alpha\, i_1(q) \big/ |h(q)|^2$$

$$\pi_1(q) = \gamma_\alpha\, i_2(q) \big/ |h(q)|^2 \quad]$$

is not known, that:

uses a suitably long block of N samples is used to establish the relationship, where N is either 4 times the number of antennae or 128, whichever is larger;

uses the result to update the receive weights at each end of the link;

optimizes the local model as in ~~EQ. 23 and EQ. 24~~ [

$$\gamma_* = \arg\min_\gamma L(\gamma, \mathbf{g}, \beta)$$

$$\gamma_\alpha = \gamma_0 + \alpha(\gamma_* - \gamma_0) \ ];$$

and then applies [

$$\pi_2(q) = \gamma_\alpha\, i_1(q) \big/ |h(q)|^2$$

$$\pi_1(q) = \gamma_\alpha\, i_2(q) \big/ |h(q)|^2 \quad]$$ ~~EQ. 25 and EQ. 26.~~

75. (currently amended) A method as in claim ~~60~~[61] that, for an aggregate proper subset m:

for each node within the set m, inherits the network objective function model given in ~~EQ. 28, EQ. 29, and EQ. 30~~ [

$$L_m(\gamma, \mathbf{g}, \beta) = \Sigma_{q \in Q(m)}\, \mathbf{g}_q\, \gamma(q)$$

$$\Sigma_{q \in Q(m)} \log(1 + \gamma(q)) \geq \beta(m)$$

$$g(q) = i_1(q) i_2(q) \big/ |h(q)|^2 \quad];$$

eliminates ~~the~~ [a] step of matrix channel estimation, transmitting instead from that node as a single real number for each link to the other end of said link an estimate of the post beamforming interference power;

and ,

receives back for each link a single real number being the transmit power.

76. (original) A method as in claim ~~75~~ [74], that for each pair of nodes assigns to the one presently possessing the most processing capability the power management computations.

77. (currently amended) A method as in claim ~~74~~[75] that estimates the transfer gains and the post beamforming interference power using simple least squares estimation techniques.

78. (currently amended) A method as in claim ~~74~~[75]that, for estimating the transfer gains and post beamforming interference power:

instead solves for the transfer gain h using ~~EQ. 31~~

[$y(n) = hgs(n) + \varepsilon(n)$];

uses a block of N samples of data to estimate h using ~~EQ. 32~~ [

$$h = \frac{\sum_{n=1}^{N} s^*(n)y(n)}{\sum_{n=1}^{N} \left|s(n)\right|^2 g}$$]

obtains an estimation of residual interference power $\cancel{R_e}$ [R_ε] using ~~EQ. 33~~ [

$$R_c = \left\langle \left|\varepsilon(n)\right|^2 \right\rangle = \frac{1}{N}\sum_{n=1}^{N}\left(\left|y(n)\right|^2 - \left|ghs(n)\right|^2\right)$$];

and,

obtains knowledge of the transmitted data symbols $s(n)$ from using remodulated symbols at the output of the codec.

79. (currently amended) A method as in claim ~~77~~ [78] wherein, instead of obtaining knowledge of the transmitted data symbols $s(n)$ from using remodulated symbols at the output of the codec, the node uses the output of a property restoral algorithm used in a blind beamforming algorithm.

80. (currently amended) A method as in claim ~~77~~ [78] wherein, instead of obtaining knowledge of the transmitted data symbols $s(n)$ from using remodulated symbols at the output of the codec, the node uses a training sequence explicitly transmitted to train beamforming weights and asset the power management algorithms.

81. (currently amended) A method as in claim ~~77~~ [78] wherein, instead of obtaining knowledge of the transmitted data symbols $s(n)$ from using remodulated symbols at the output of the codec, the node uses any combination of:

the output of a property restoral algorithm used in a blind beamforming algorithm;

a training sequence explicitly transmitted to train beamforming weights and asset the power management algorithms;

or,

other means known to the art.

82. (currently amended) A method as in claim ~~60~~[61], wherein each node incorporates a link level optimizer and a decision algorithm~~, as illustrated in Figure 32Aand 32B~~.

83. (currently amended) A method as in claim ~~81~~[82], wherein the decision algorithm is a Lagrange multiplier technique.

84. (currently amended) A method as in claim ~~60~~[61], wherein the solution to ~~EQ. 3~~ [$\min_{\pi_1(q)} \sum_q \pi_1(q) = 1^T \pi_1$] is implemented by a penalty function technique.

85. (currently amended) A method as in claim ~~83~~[84], wherein the penalty function technique:

takes the derivative of $\cancel{\gamma_{(q)}}$ [$\gamma(q)$] with respect to π_1;

and,

uses the Kronecker-Delta function and the weighted background noise.

86. (currently amended) A method as in claim ~~83~~[84], wherein the penalty function technique neglects the noise term.

87. (currently amended) A method as in claim ~~83~~[84], wherein the penalty function technique normalizes the noise term to one.

88. (currently amended) A method as in claim ~~60~~[61], wherein the approximation uses the receive weights.

89. (currently amended) A method as in claim ~~60~~[61], wherein adaptation to the target objective is performed in a series of measured and quantized descent and ascent steps.

90. (currently amended) A method as in claim ~~60~~[61], wherein the adaptation to the target objective is performed in response to information stating the vector of change.

91. (currently amended) A method as in claim ~~60~~[61], which uses the log linear mode ~~in EQ. 34~~ [

$$\beta_q \approx \log\left(\frac{a\ \ \pi_1(q)+a_0}{b\ \ \pi_1(q)+b_0}\right)=\hat{\beta}_q\left(\pi_1(q)\right)$$]

and the inequality characterization ~~in EQ. 35~~ [$\hat{\beta}_q(\pi_1(q))\geq\beta$] to solve the approximation problem with a simple low dimensional linear program.

92. (currently amended) A method as in claim ~~60~~[61], develops the local mode by matching function values and gradients between the current model and the actual function.

93. (currently amended) A method as in claim ~~60~~[61], which develops the model as a solution to the least squares fit, evaluated over several points.